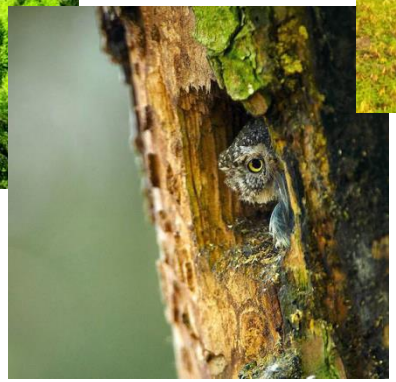
A multicriteria cutting algorithm To simulate innovative silvicultures

Benoit Courbaud
Gilles Le Moguedec
François De Coligny

Journées CAQSIS 27-29 April 2018 Clermont Ferrand

Development of a variety of complex silvicultures

- Crown thinning
- Uneven-aged stands
- Retention forestry
- Gap creation
- Mixed stands
- Seed tree selection



Fores dynamics models need tools to simulate these silvicultures

Tradition in silviculture experimentation:
classification of thinning operations
(Assmann 1961, 1970; Abetz & Mitscherlich 1969; Schütz 1990, Pretzsch 2009)

Three axes

- kind of thining (low, crown, selection, geometric …)
- severity of thinning (tree nb, basal area, stand density index …)
- rotation (age of first thinning, tree size at thinning …)

Restricted to thinning
Restricted to even-aged silviculture
Structure centered
Segregation in small boxes but lack of unifying concept

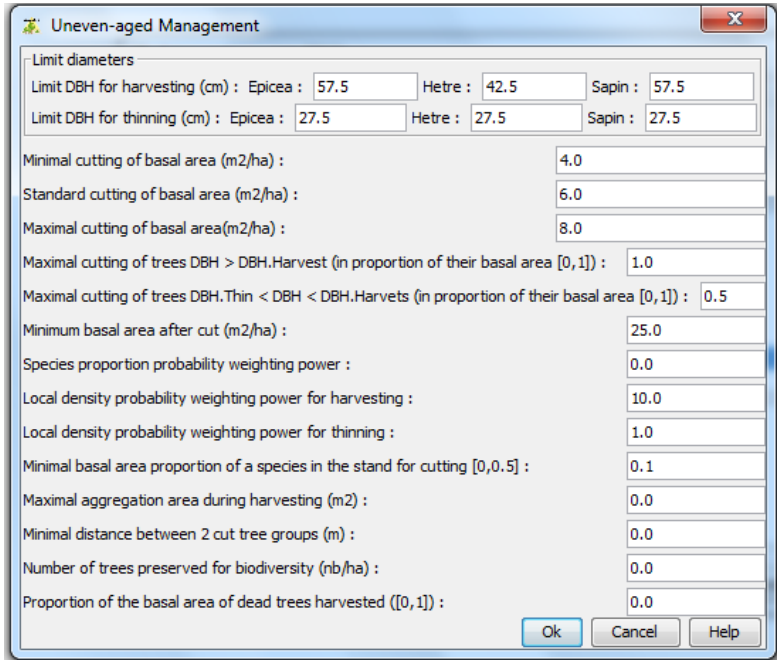
Tradition in uneven-aged silviculture:
The selection cutting combines multiple objectives : harvest, thin, regenerate …

ProSilva method:
Determine allowable cut (allowable annual cut volume)
Decide to retain or cut each individual tree considering
- A list of reasons to retain a tree
- A list of reasons to cut a tree

A gradient of individual preferences appears among silviculturists

Simulation approaches

Capsis uneven-aged manager
(Lafond et al., 2014, 2015, 2017
Courbaud et al., 2017)



A cutting strategy is defined
by a set of parameters

- Combines multiple criteria
- Virtual experiments in the parameter spaces
- Sensitivity of simulation results to silviculture parameters
- Silviculture optimisation

Limits
- Difficulty to understand the role of each parameter
- Difficulty to evolve: accumulation of ad-hoc constraints
- Unclear silviculture strategy

Objective

Build a new cutting algorithm that makes possible to :

- Simulate a large variety of strategies with a single tool
- Combine multiple criteria
- Give an equivalent role to retention and cutting
- Add new criteria easily
- Unify cutting strategies as points in a parameter space with a clear role for each parameter

Ideas

- Double selection : Retention vs. Cutting
- Use a multicriteria optimisation approach (Kangas)
- Rank trees regarding retention and cutting criteria

Principle of the multicriteria thinner

Optional Step 1: Select a population of focal trees = structuring trees

Determine a global target for Focal trees (e.g. N)
Rank trees following a focal multicriteria index
Select trees in order of decreasing index up to the target

Step 2: Select a population of cut trees

Determine a global target for Cut trees (e.g. G after cut)
Allow or not to cut focal trees
Rank trees following a cutting multicriteria index
Select trees in order of decreasing index up to the target

Rq: Double optimization:

- The cumulated focal index of focal trees is maximized
- The cumulated cutting index of cut trees is maximized

Building an individual tree multicriteria index:
- Linear combination of individual tree criteria
- Weight of each criteria defined by the user

Example:

Focal_Index = wf1*Species
+ wf2*DBH
+ wf3 * Nb_tree_microhabitats
+ wf4 *Distance_to_nearest_focal_tree

Cut_Index = wc1*Species
+ wc2*DBH
+ wc3 *Distance_to_nearest_focal_tree
+ wc4 *Distance_to_nearest_cut_tree

Technical difficulty with distances:
Distances among trees of a same group (focal or cut) cannot be calculated before selecting the group

Optimisation approach by simulated annealing

Select a first group of N potential focal trees
Calculate **Global_Score** = sum of individual tree focal indexes

Select a new potential focal tree
Replace the worst potential focal tree by the new potential focal tree
Calculate **New_Global_Score**

If **New_Global_Score > Global_Score**
Accept the new focal tree population
If **New_Global_Score < Global_Score**
Accept the new focal tree population with probability p
Reject the new focal tree population with probability (1-p)

Iterate

Simulated annealing:

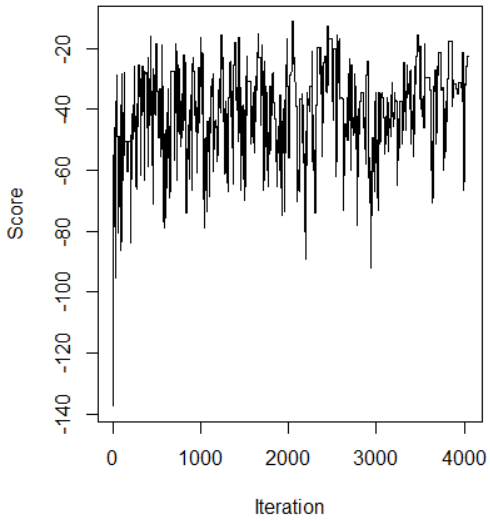
The probability of acceptance p diminishes

when the new score is lower than the previous score

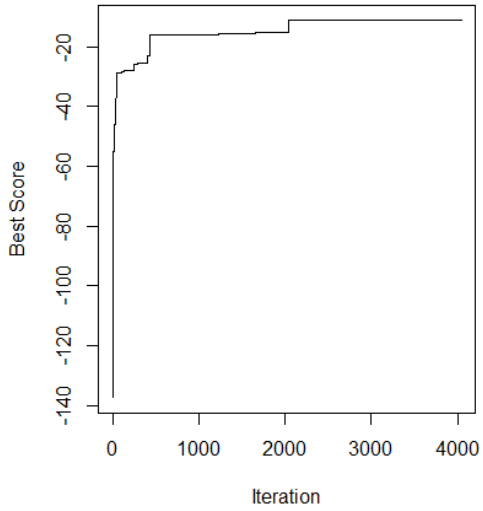
when the number of iterations increases

Accepting sometimes focal tree populations with low scores makes possible to explore widely the space of possibilities
And avoids beeing trapped in a local maximum

The focal tree population with the best score is memorized



Evolution of Global_Score along iterations



Evolution of Best_Global_Score along iterations

Capsis interface

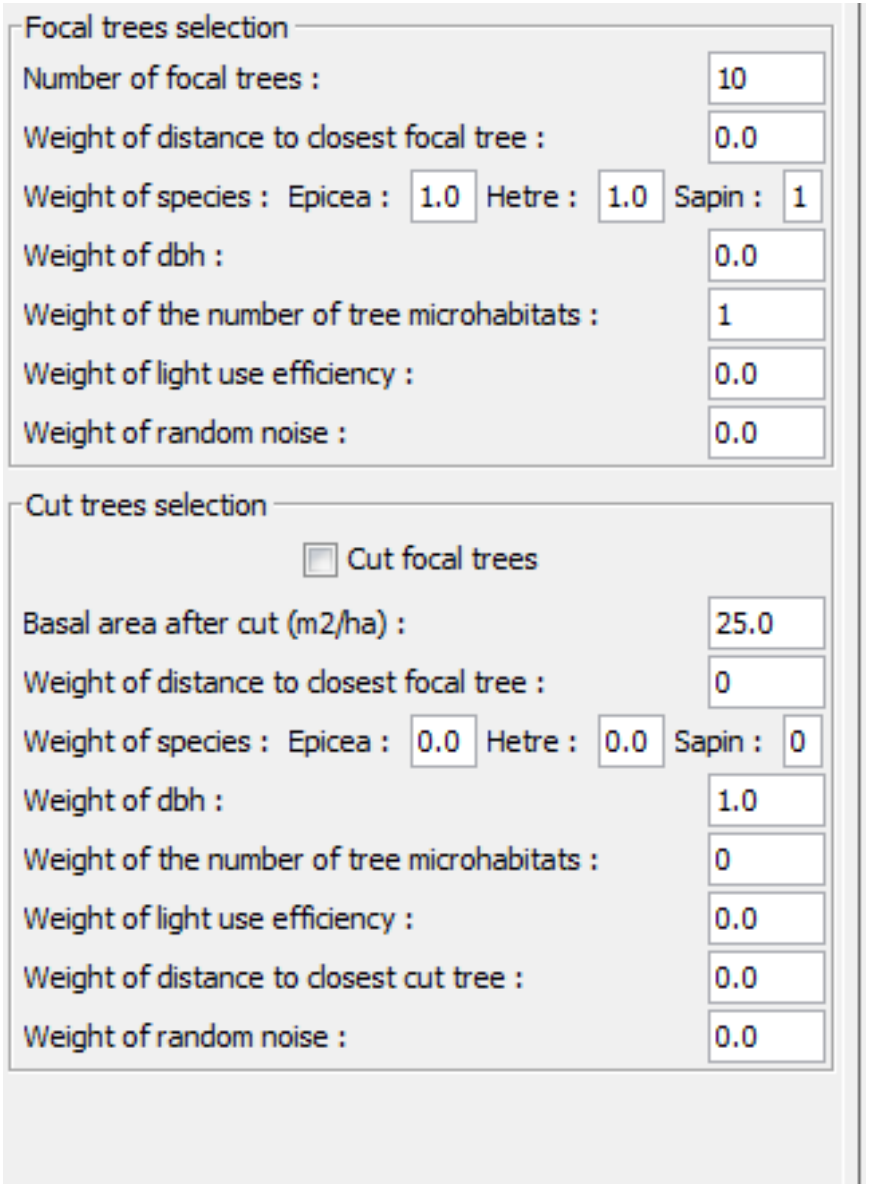
Focal quantity Target

Weights of focal criteria

Cut focal trees : yes/no

Cuting quantity target

Weights of cuting criteria

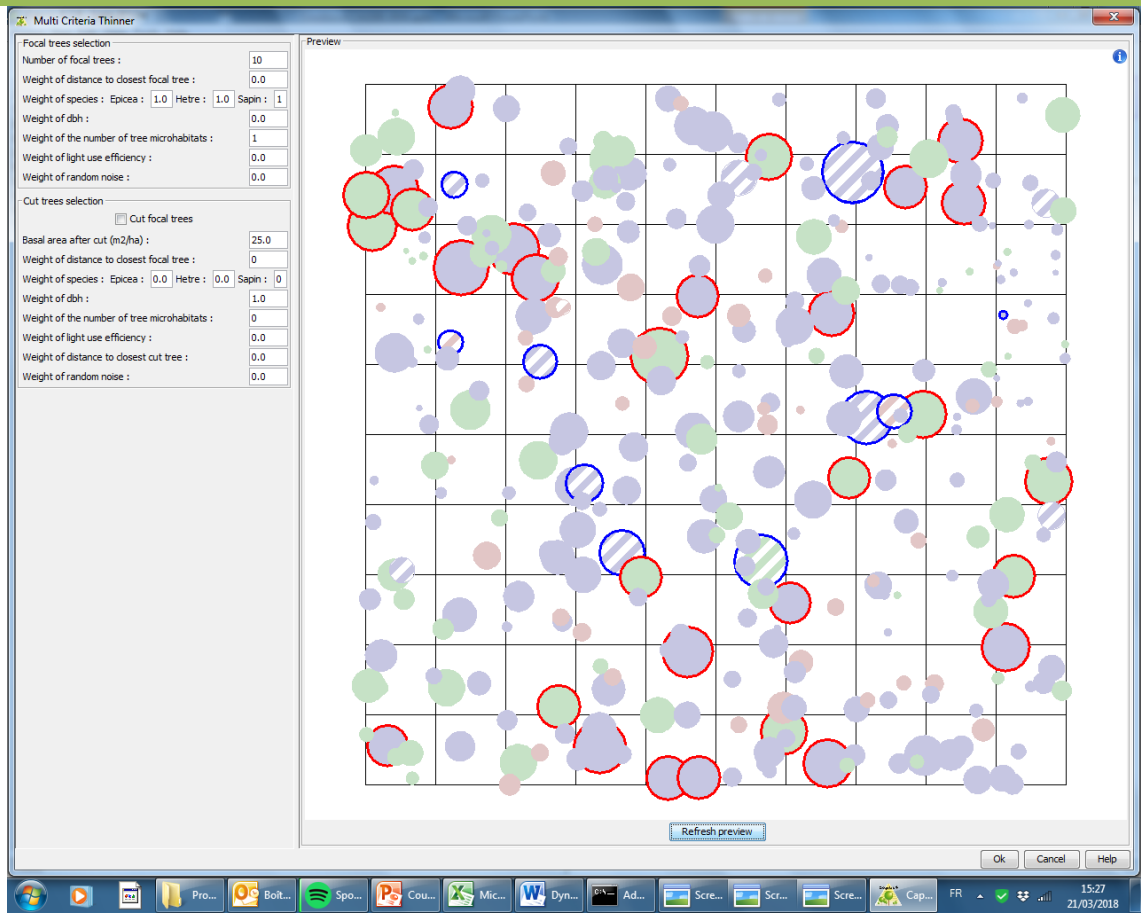


Results

Retain habitat trees

Cut huge trees

Focal trees in blue
Cut trees in red



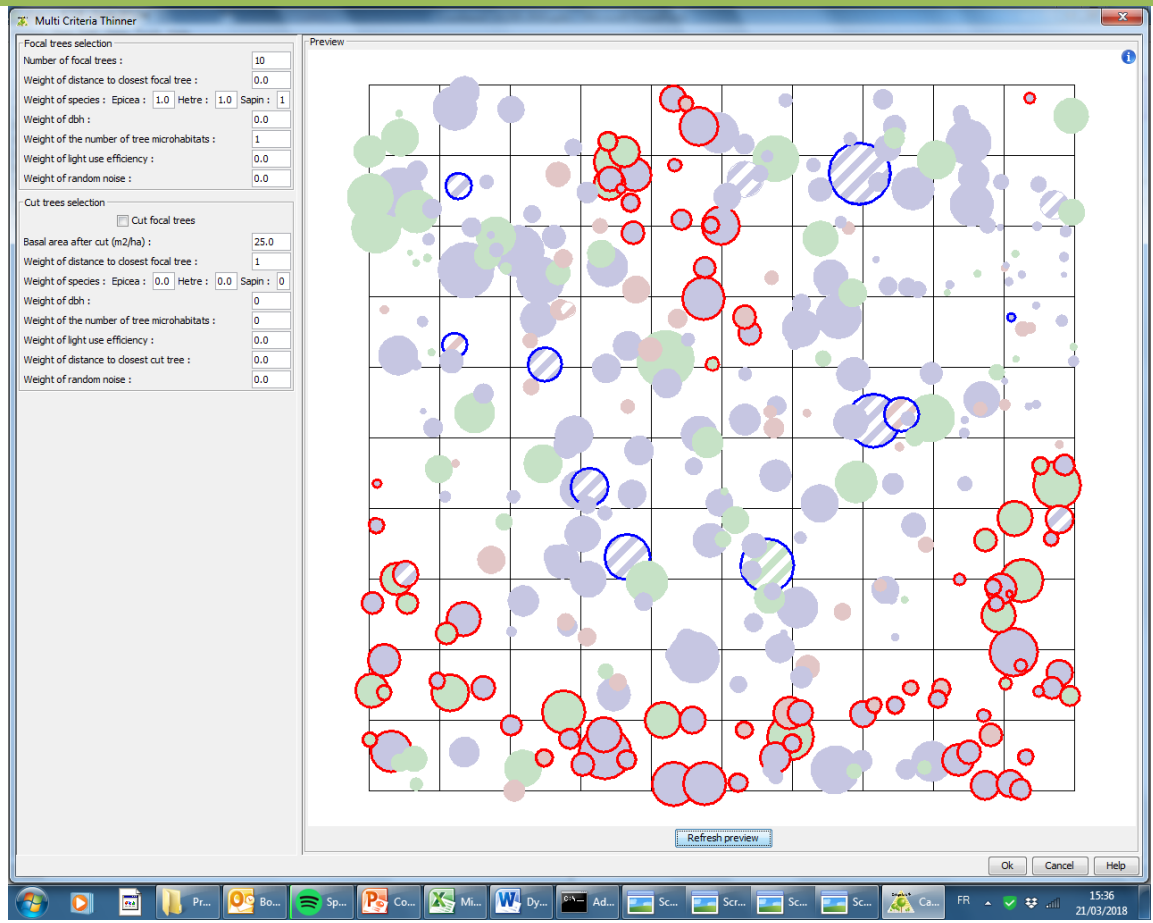
Focal trees
N=10 Focal_Index = 1 * Nb_tree_microhabitats

Cut trees
G_AfterCut = 25 Cut_Index = 1 * Dbh

Results

Cut trees
Far from focal trees

(repulsion)



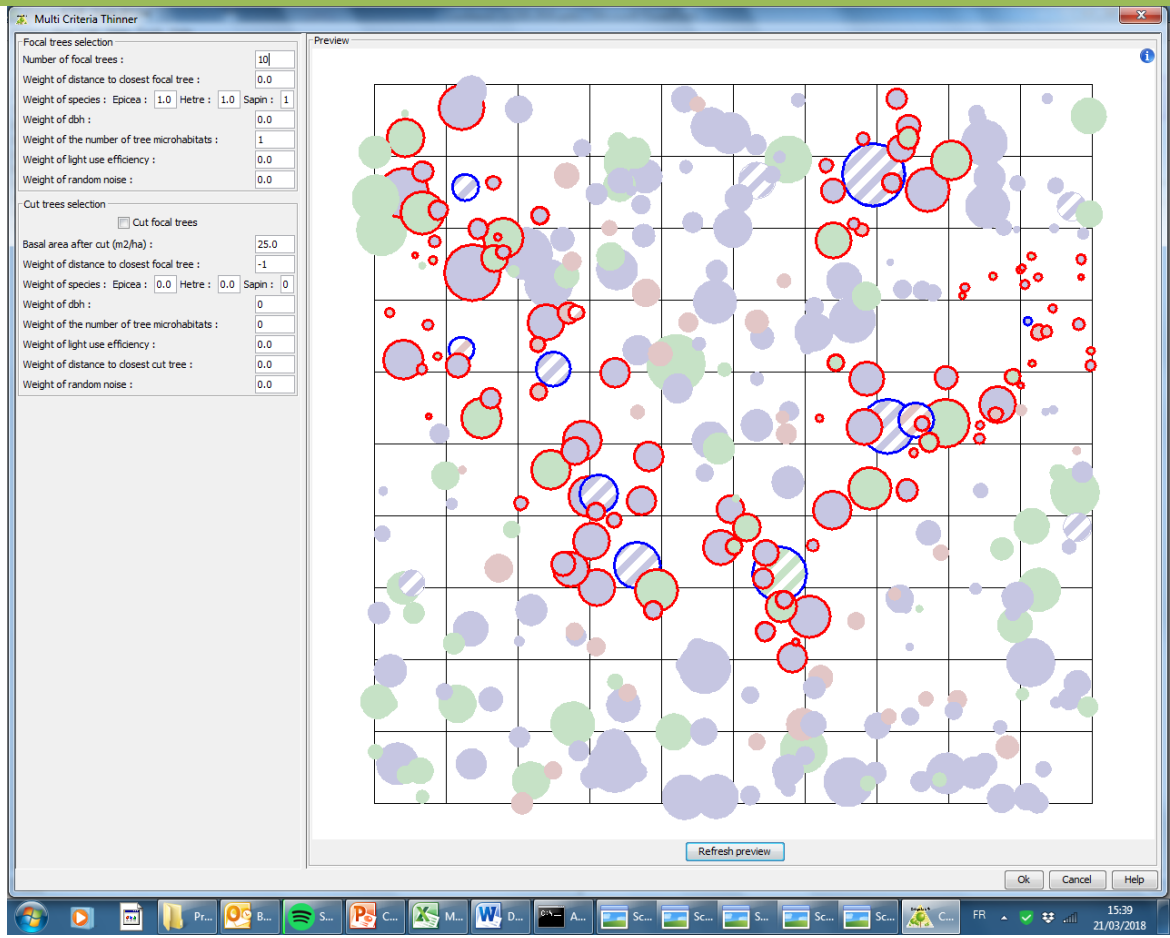
Focal trees
N=10 Focal_Index = 1 * Nb_tree_microhabitats

Cut trees
G_AfterCut = 25 Cut_Index = 1 * Distance_to_nearest_focal_tree

Results

Cut trees
Close to focal trees

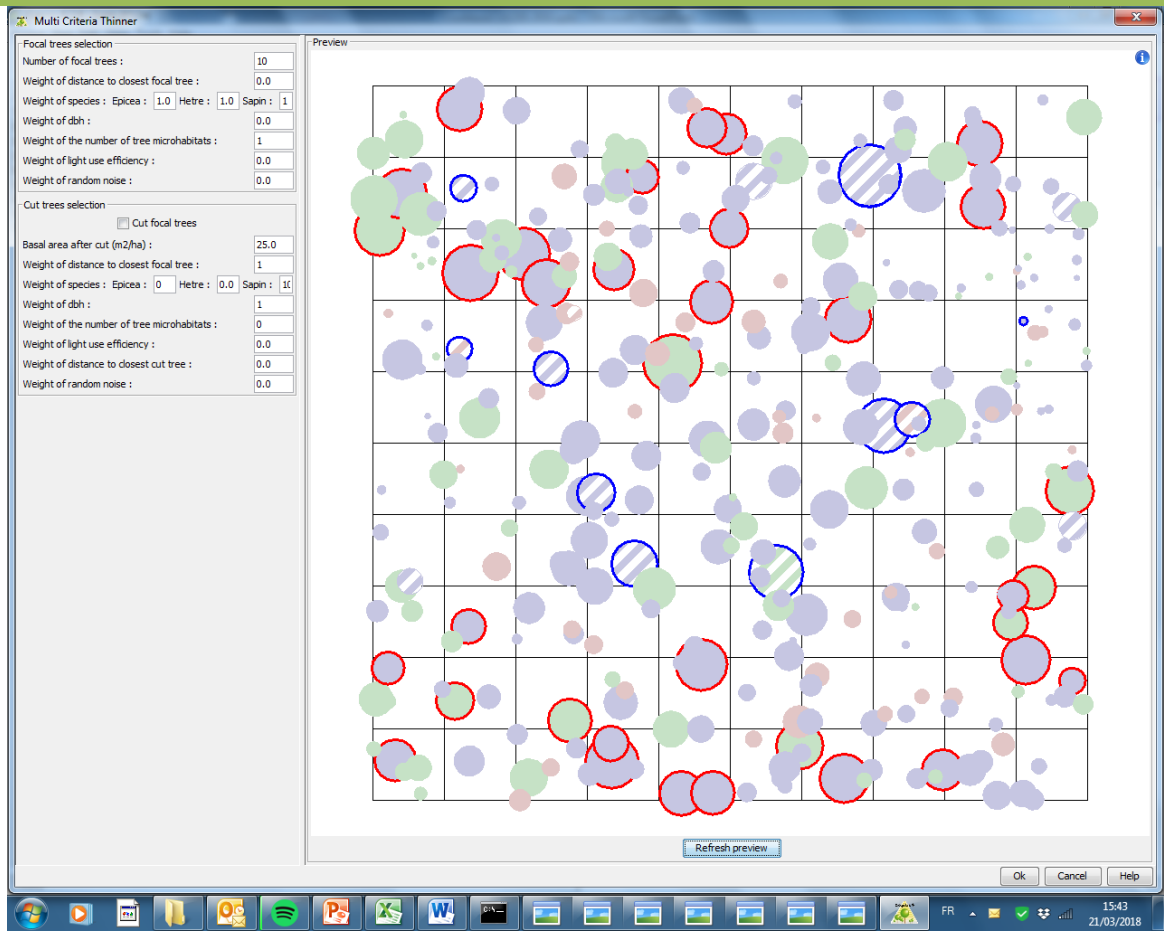
(Attraction)



Focal trees
N=10 Focal_Index = 1 * Nb_tree_microhabitats

Cut trees
G_AfterCut = 25 Cut_Index = -1 * Distance_to_nearest_focal_tree

Cut
Huge
Fir trees
Distant from focal trees



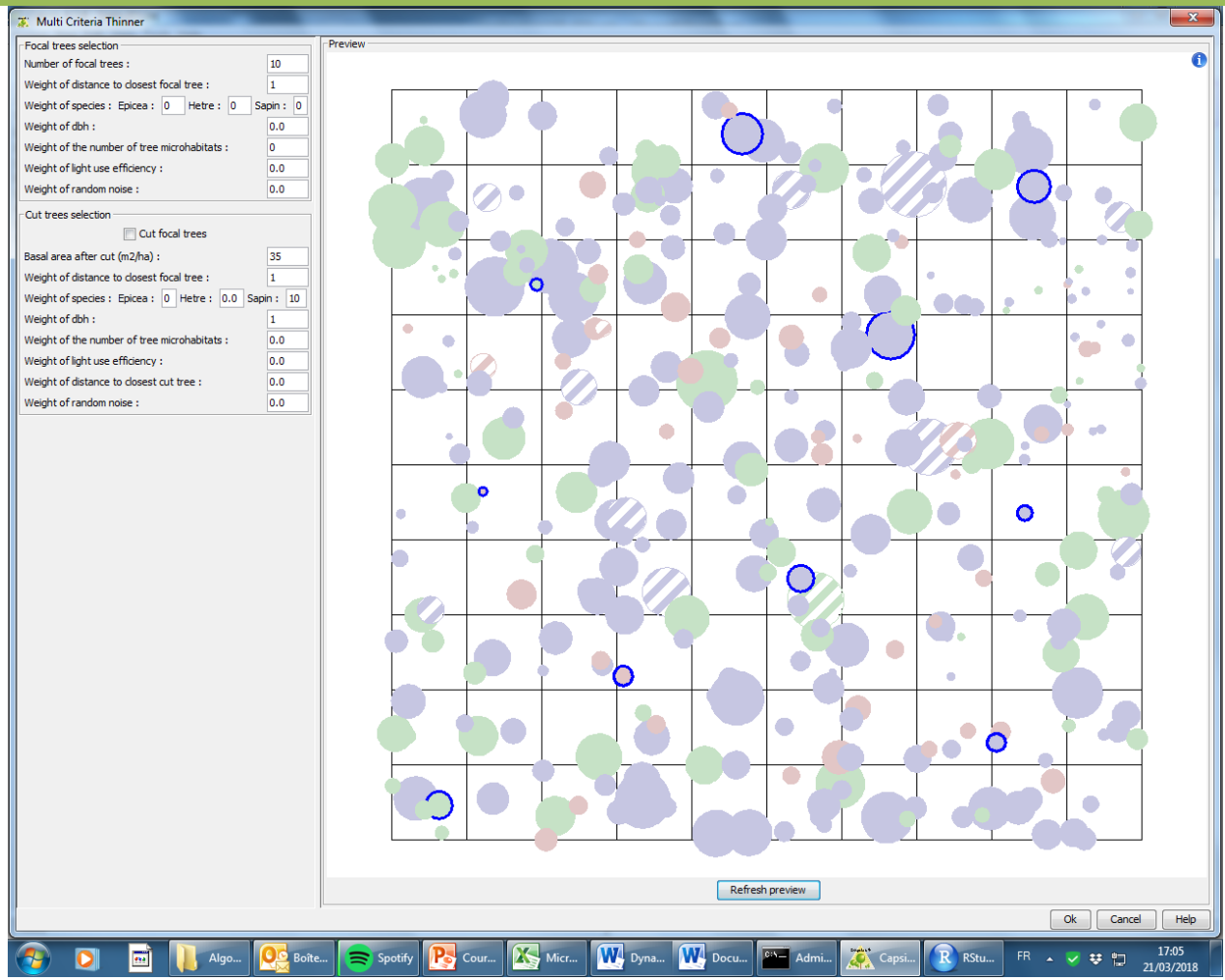
Focal trees
N=10 Focal_Index = 1 * Nb_tree_microhabitats

Cut trees
G_AfterCut = 25 Cut_Index = 10 * Fir + 1* DBH + 1 * Distance_to_nearest_focal_tree

Results

Repulsion
between
Focal trees

Distances
calculated
In a torus system



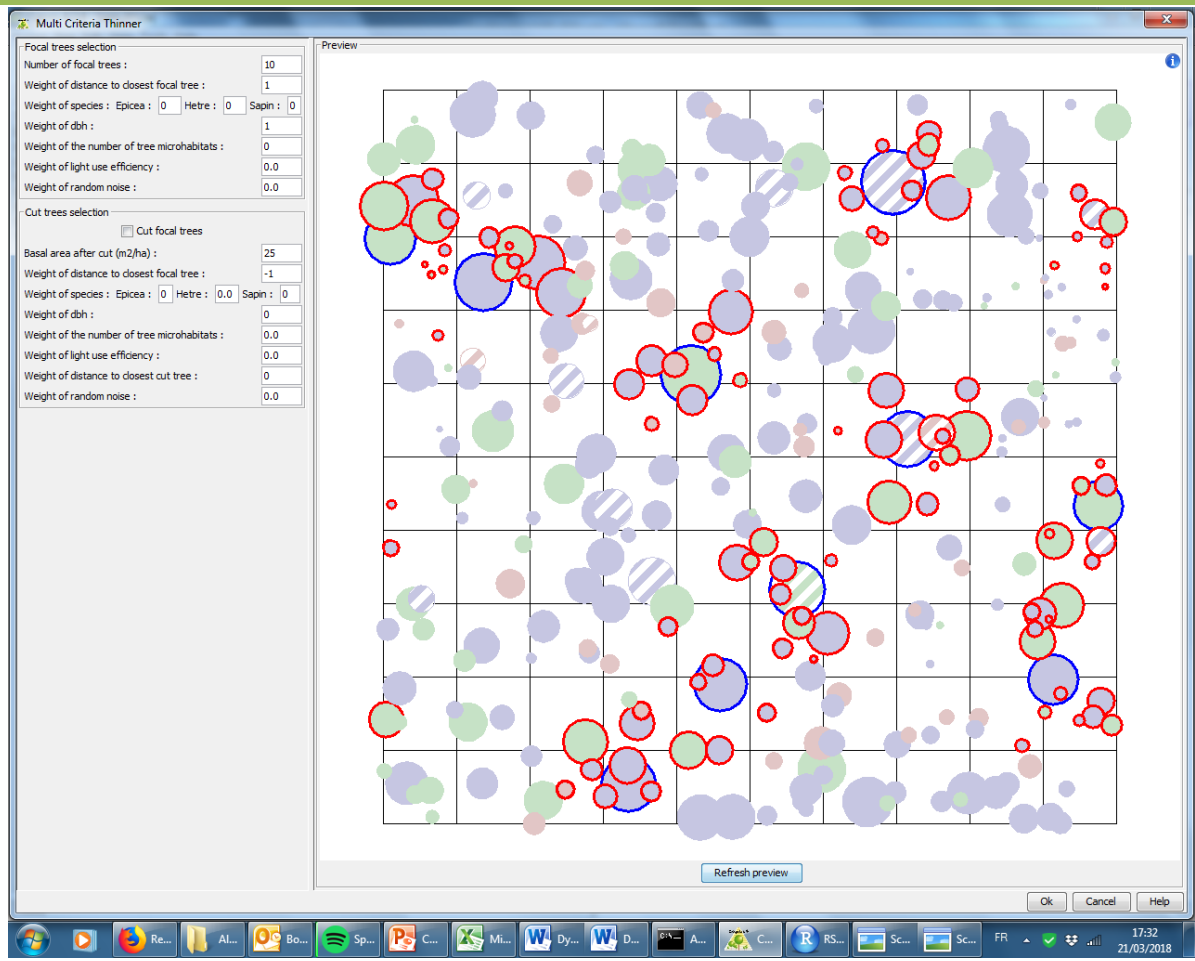
Focal trees

N=10 Focal_Index = 1 * Dist_to_nearest_focal _tree

Results

Focal trees
Huge
Distant from other
Focal trees

Cut
Trees close to
Focal trees



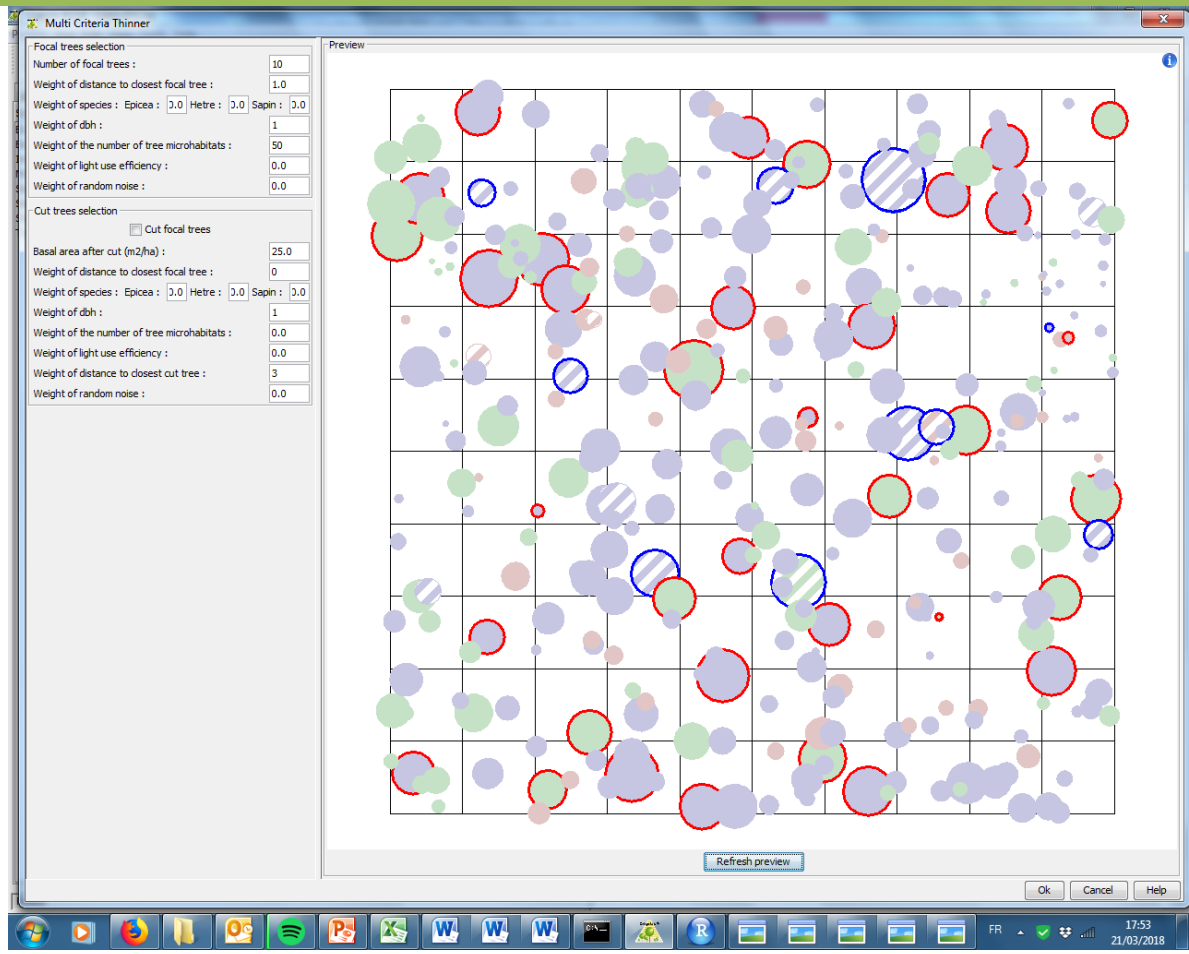
Focal trees
N=10 Focal_Index = 1 * DBH + 1* Dist_to_closest_focal

Cut trees
G_AfterCut = 25 Cut_Index = -1 * Distance_to_nearest_focal_tree

Results

Retain
Huge
Habitat trees

Cut
Huge,
Trees distant from
Other cut trees



Focal trees

N=10 Focal_Index = 50 * Nb_tree_microhabitats + 1 * DBH

Cut trees

G_AfterCut = 25 Cut_Index = 1* DBH + 3 * Distance_to_nearest_cut_tree

Discussion

Benefits:

Possibility to reproduce a high variety of strategies

New criteria can be added easily
Randomness can be added as a new criteria

Fast computing if a distance matrix is calculated only once

Qualitative and quantitative criteria can be combined

Good framework to
formalize expert strategies
do simulation experiments

Limits:

Numerical optimisation -> a same set of parameters can produce different tree selections

Perspectives

Test the tool !

Standardize criteria to make weighting easier

Introduce non-linearities to represent preferences among criteria better

Add new criteria

Compare strategies making simulation experiments in the weight space

Estimate expert strategies on marteloscopes


Thank you for your attention